OCR
Oxford Cambridge and RSA

An OCR endorsed teaching and learning tool

OCR A Level Computer Science

H446 – Paper 1

1

Functions of an Operating System

Unit 2

Systems software and applications generation

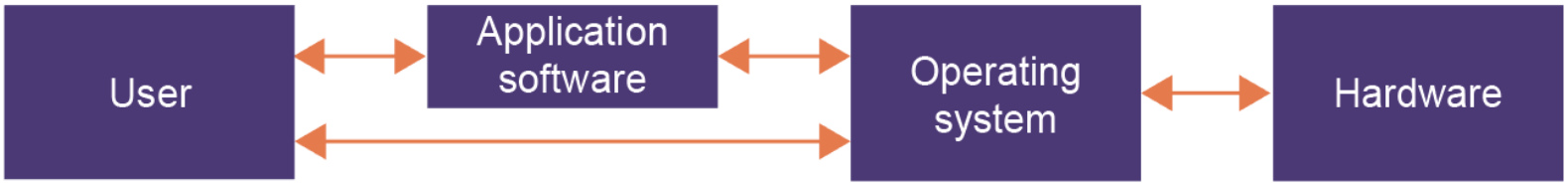
PG ONLINE

Objectives

- Understand the function and purpose of an operating system
- Describe memory management (paging, segmentation and virtual memory)
- Describe the role of interrupts and an Interrupt Service Routine (ISR) within the fetch-decode-execute cycle
- Describe the need for processor scheduling algorithms
- Describe scheduling algorithms: round robin, first come first served, multi-level feedback queues, shortest job first and shortest remaining time

What is an Operating System?

- You need software to manage communication with your computer hardware
- The boot loader in ROM loads the Operating System (OS) into RAM when the computer is switched on
- The OS manages the hardware and provides an interface for the user and the application

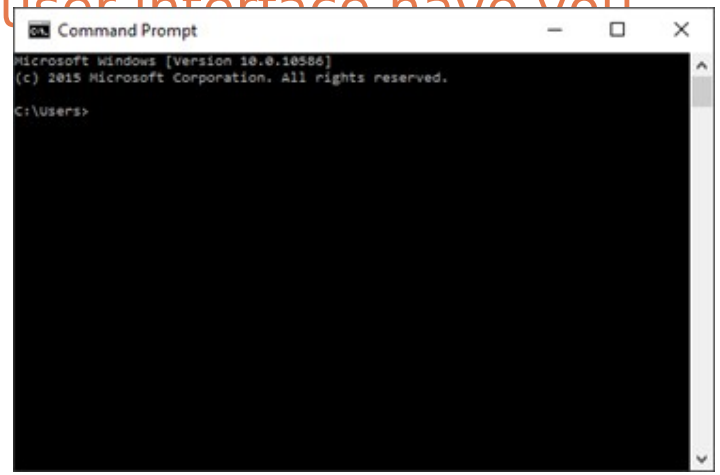


What functions does the OS provide?

- User interface
- Memory management
- Interrupt handling
- Processor scheduling

User interface

- The Operating System hides the complexity of the hardware from the user by providing a user interface
  - Other than a desktop computer, what devices might have an operating system?
  - What different types of OS user interface have you



Memory management

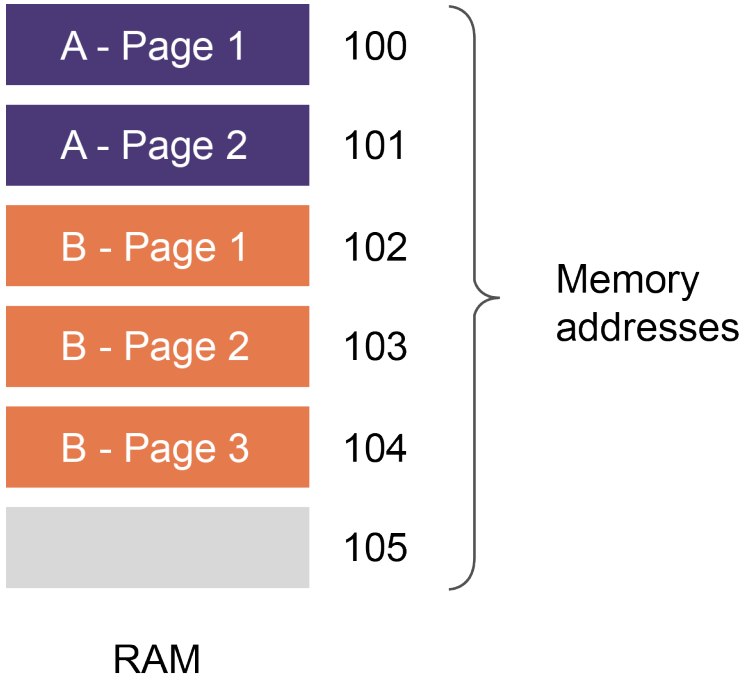
- Programs and their data need to be loaded into RAM
- The Operating System must manage the allocation of RAM to the different programs
- There may not be sufficient RAM for all desired processes to be completely loaded into RAM at once

Paging

- Available memory is divided into fixed size chunks called pages
- Each page has an address
- A process loaded into RAM is allocated sufficient pages, but those pages may not be contiguous (next to each other) in physical terms

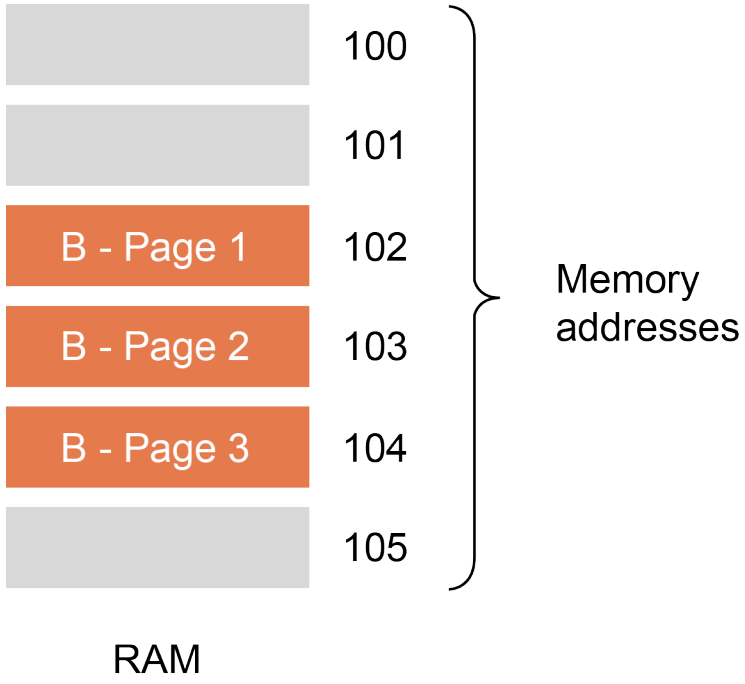
Paging

1. Process A requires two pages in RAM, process B requires three pages



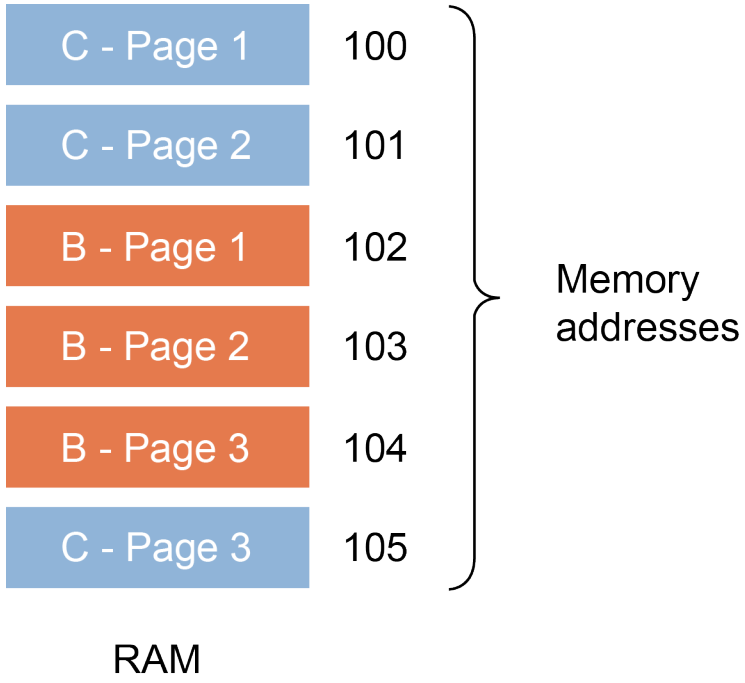
Paging

2. Process A ends



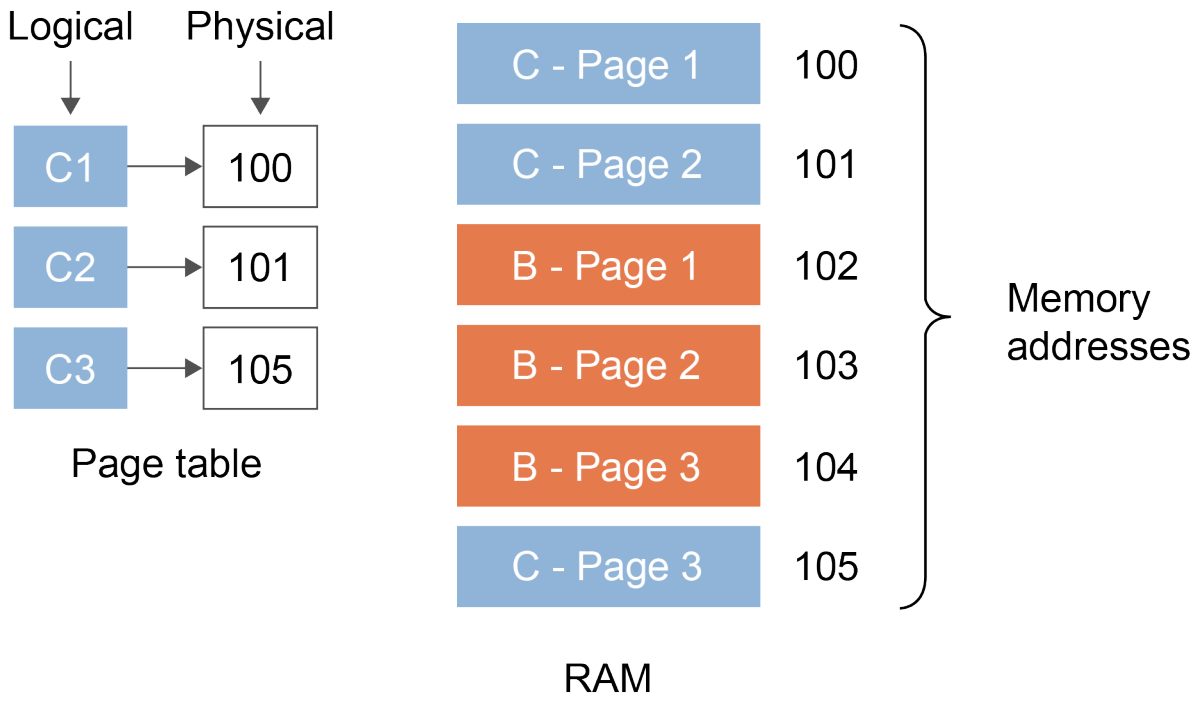
Paging

3. Process C is started, but it needs three pages of RAM so it is allocated non-contiguous pages



Page table

A page table maps between the logical memory locations and the physical memory locations



Segmentation

- Alternatively, memory is divided into segments which can be of different lengths
- Segments can relate to parts of a program, for example a particular function or subroutine may occupy a segment

Virtual memory

- A computer has a fixed amount of RAM; the demands for memory will often exceed this amount
- An area of the hard disk can be designated as virtual memory
- Some of the pages of a current process are stored in virtual memory until they are needed, at which point they are swapped into RAM

Virtual memory

- If many processes are running and the computer has insufficient RAM, lots of time is spent swapping pages in and out of virtual memory
- Repeatedly swapping pages can noticeably slow down the computer
- This is known as disk thrashing

Worksheet 1

Complete the questions in **Task 1**

Interrupts

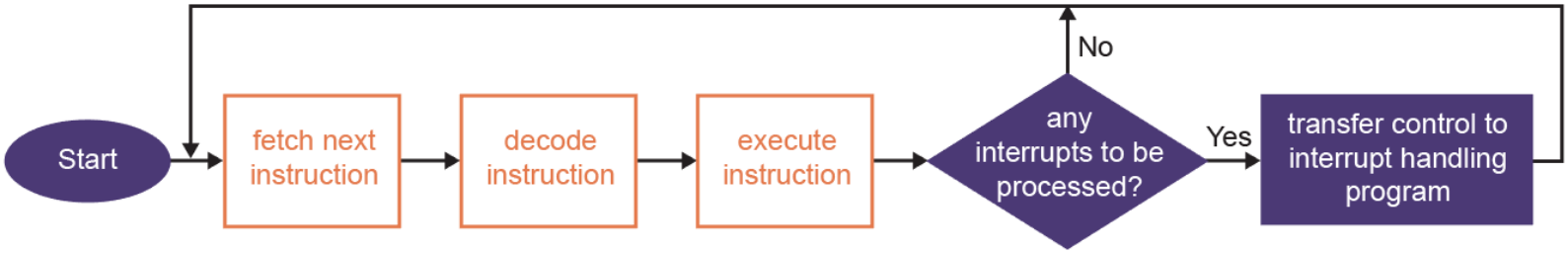
- It is vital that the CPU can be interrupted when necessary
- Interrupts can be sent to the CPU by software, hardware devices or the CPU's internal clock
  - Can you think of any reasons why processing might need to be interrupted?

Interrupt examples

- An I/O device sends an interrupt signal
- The printer runs out of paper
- An error occurs in a program
- A scheduled interrupt from the internal clock
- Power failure

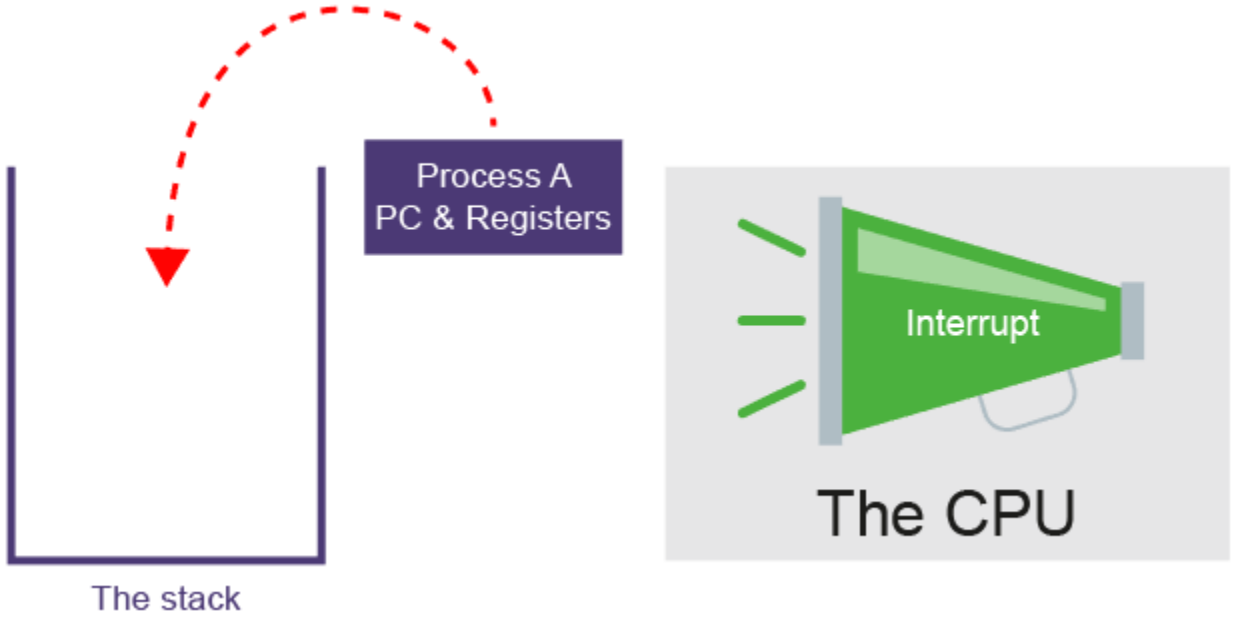
Interrupts

- The CPU checks at the end of each clock cycle whether there are any interrupts to be processed



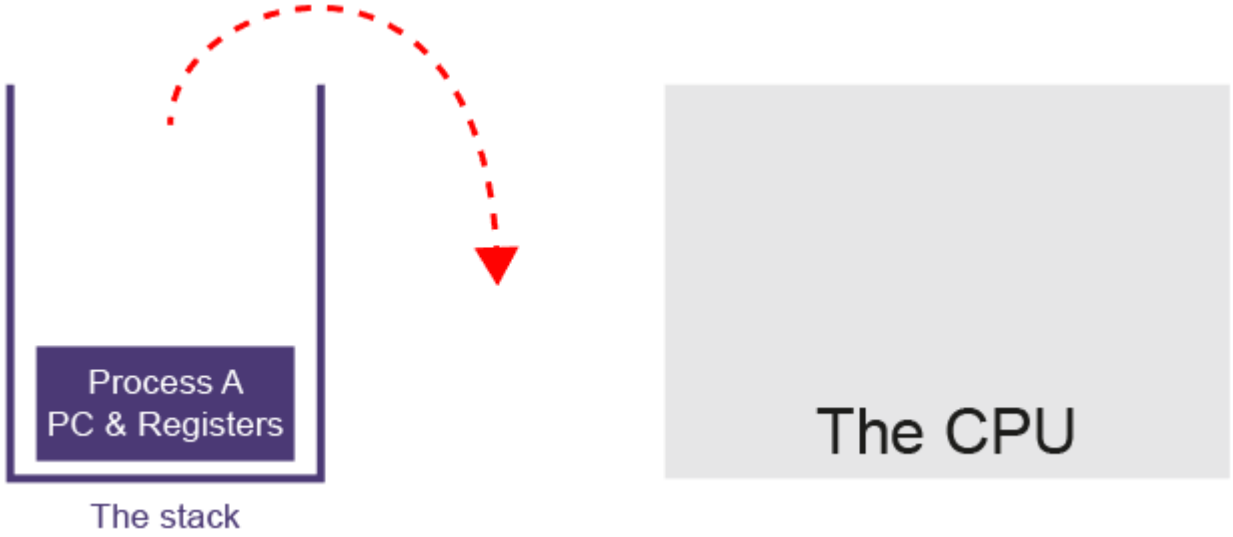
Interrupts – using the stack

- When an interrupt is detected, the processor stops fetching instructions and instead pushes the current contents of its registers onto a stack



Interrupts – using the stack

- The CPU uses an interrupt service routine to process the interrupt
  - When processing has finished, the values can be popped from the stack and reloaded into the CPU

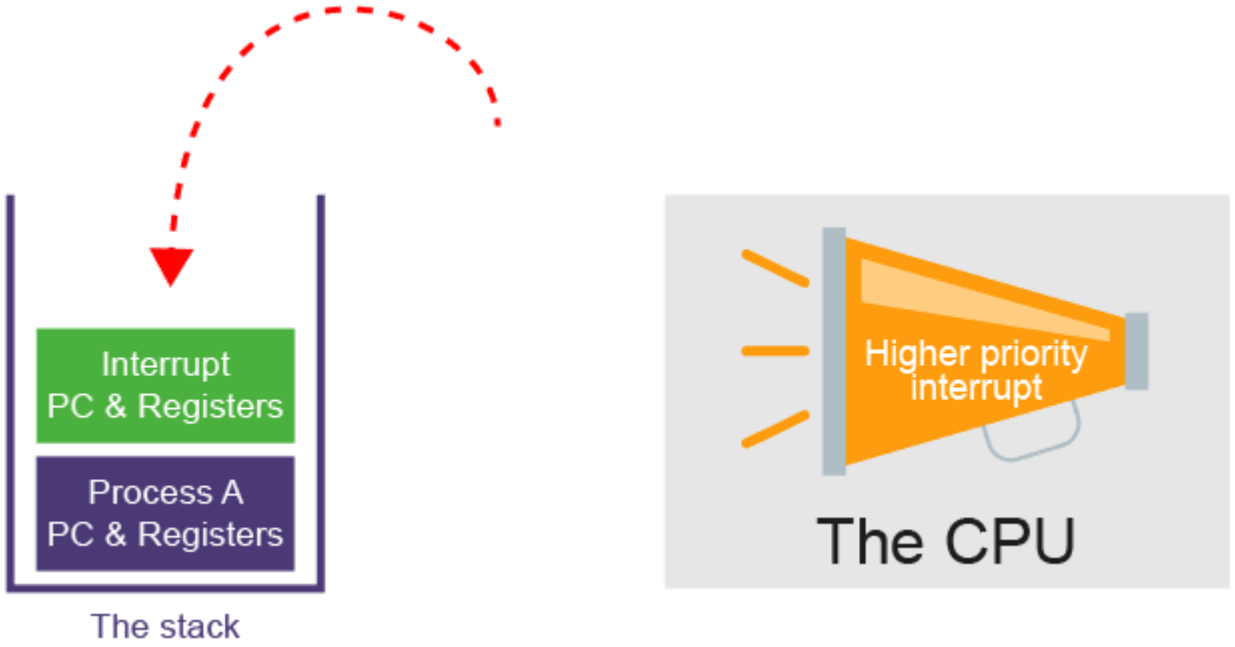


Interrupt priority

- Interrupts have different priorities, and will be processed in order of priority
- Interrupts can themselves be interrupted if the new interrupt is of a higher priority

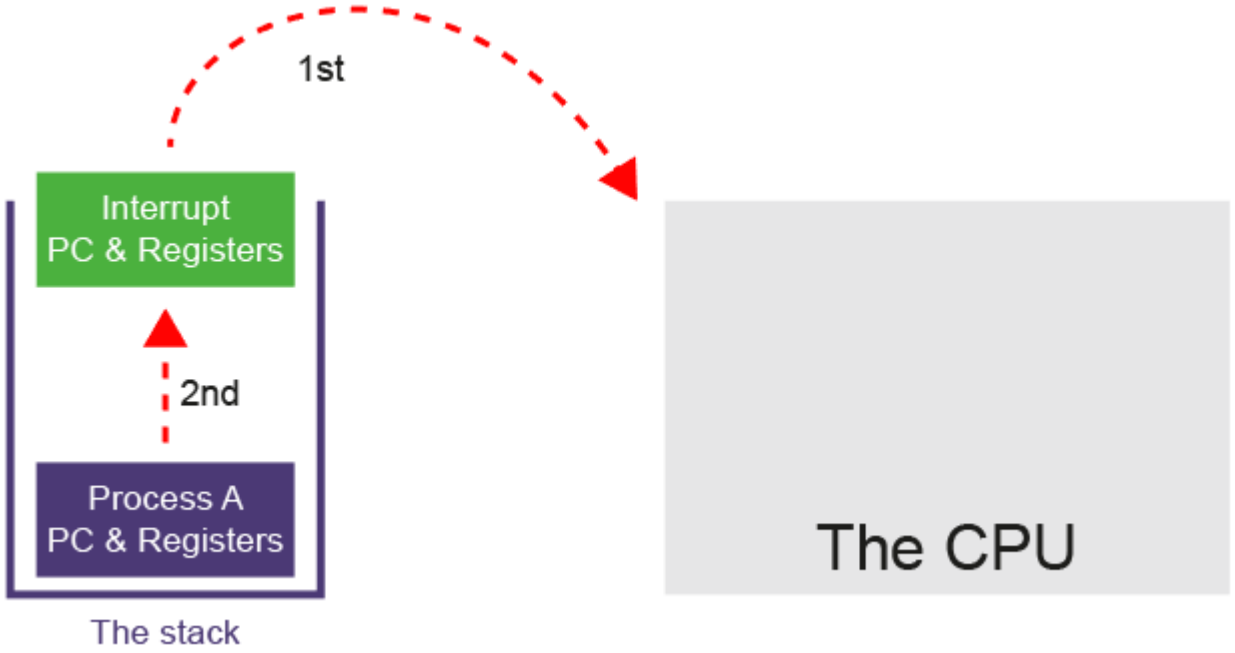
Interrupt priority

- If a higher priority interrupt occurs whilst an interrupt is being processed, the original interrupt's registers will be pushed onto the stack as well



Interrupt priority

- A stack is a LIFO data structure, so the last data to be pushed on will be the first to be retrieved



Processor scheduling

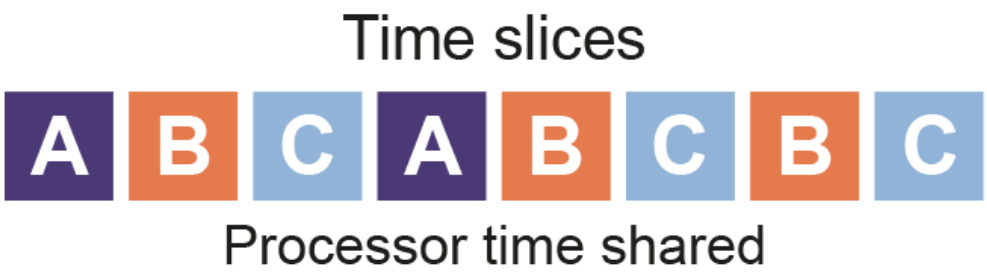
- A single CPU can only process instructions for one application at a time
- The Operating System must schedule when each application can use the CPU
- This gives the illusion of multi-tasking – multiple applications appear to be running simultaneously

Aims of scheduling

- To provide an acceptable response time to all users
- To maximise the time the CPU is usefully engaged
- To ensure fairness on a multi-user system

Round Robin

- Each job is allocated (by FIFO) a time slice during which it can use the CPU's resources
  - How does the scheduler know when to switch between time slices?
- If the job has not completed by the end of its time slice, the next job is allocated a time slice



First come first served

- The first job to arrive is executed until it completes
  - What do you think the drawbacks of this system might be?

Shortest remaining time

- The time to completion is estimated as each new
  job arrives
- The job with the shortest remaining time to completion is executed, meaning that a shorter new job can take over from the current process
  - Why does this algorithm not constantly switch between two jobs as they become closer to completion?
  - What do you think ‘starvation’ is, and why might it occur if this algorithm is used to schedule processor time?

Shortest job first

- Also known as “shortest process next”
- As with shortest remaining time, the total execution time of each job is estimated by the user
- The waiting job with the smallest total execution time is executed when the current job completes
  - Unlike shortest remaining time, this algorithm is not pre-emptive. What do you think this means?

Multi level feedback queues

- Multiple queues are created with different priority levels
- If a job uses too much CPU time it is moved to a lower priority queue
- Processes can also be moved to a higher priority queue if they have waited a long time

Worksheet 1

- In pairs or groups, complete **Task 2**

Plenary

- The Operating System (OS) manages memory allocation, CPU time and provides an interface
- Memory can be allocated in pages
  - Virtual memory allows pages to be swapped in and out of RAM as they are needed
- The CPU processes jobs according to a scheduling algorithm
- The CPU can be interrupted if a job with a higher priority arrives

Copyright

Licence agreement

PG ONLINE